In [1]:
```python
#1. Task: Calculate Profit Percentage


#• Write a javascript program that takes input for the cost price and selling price
#an item.
#• Hints
#o Prompt the user to input the cost price and selling price.
#o Determine whether the transaction resulted in a profit or loss.
#o If there is a profit calculate the profit percentage; if there is a loss
#calculate the loss percentage.
#o Display the profit or loss and the respective percentage.



cost_price = float(input("Enter the Cost Price (₹): "))
selling_price = float(input("Enter the Selling Price (₹): "))
if selling_price > cost_price:
    profit = selling_price - cost_price
    profit_percent = (profit / cost_price) * 100
    print(f"Profit of ₹{profit:.2f}")
    print(f"Profit Percentage: {profit_percent:.2f}%")
elif cost_price > selling_price:
    loss = cost_price - selling_price
    loss_percent = (loss / cost_price) * 100
    print(f"Loss of ₹{loss:.2f}")
    print(f"Loss Percentage: {loss_percent:.2f}%")
else:
    print("No Profit, No Loss.")
```

```
Profit of ₹200.00
Profit Percentage: 100.00%
```

In [15]:
```python
#2. Task: Cricket Stats Analyzer
#• Objective: Write a script to analyze cricket stats for a team.
#• Hints:
#o Prompt the user to input the runs scored by each of the five players in a
#o For each player (Player 1 to Player 5) ask the user to input the runs they
#scored.
#o Calculate the total runs scored by all players and the average runs.
#o Display the total runs and average runs to the user.



runs = []
for i in range(1, 6):
    score = int(input(f"Enter runs scored by Player {i}: "))
    runs.append(score)
total_runs = sum(runs)
average_runs = total_runs / len(runs)
print(f"\n Total Runs Scored by Team: {total_runs}")
print(f" Average Runs per Player: {average_runs:.2f}")
```

```
Total Runs Scored by Team: 200
Average Runs per Player: 40.00
```

In [3]:
```python
#3. Task: Retirement Age Calculator
#● Objective: Write a program that prompts the user for their age and tells them ho
#many years until they reach retirement age (65).
#● Hints:
#○ Ask the user to input their age.
#○ Calculate how many more years they have until they reach 65 years of
#age.
#○ Display the number of years left until retirement or a message if the user
# already reached retirement age.



age = int(input("Enter your current age: "))
retirement_age = 65
if age < retirement_age:
    years_left = retirement_age - age
    print(f"You have {years_left} years left until retirement at age {retirement_ag
else:
    print("🎉 You have already reached retirement age!")
```

You have 38 years left until retirement at age 65.

In [11]:
```python
#4. Task: Calculate the Area of a Circle
#● Objective: Write a program to calculate the area of a circle.
#● Hints:
#○ Ask the user to input the radius of the circle.
#○ Calculate the area of the circle using the formula: Area = π *
#radius^2.
#○ Display the calculated area.



import math
radius = float(input("Enter the radius of the circle (in cm): "))
area = math.pi * radius ** 2
print(f" Area of the Circle: {area:.2f} square cm")
```

Area of the Circle: 78.54 square cm

In [12]:
```python
#5. Task: Salary Calculation
#● Objective: You have to calculate an employee's salary by computing the gross
#salary tax and net salary based on the given parameters.
#● Hints:
#○ Base Salary = ₹50000
#○ Bonus = ₹5000
#○ Tax Rate = 10%
#○ Other Charges = ₹2000
#Display the Gross Salary Tax and Net Salary.



base_salary = 50000
bonus = 5000
tax_rate = 0.10
other_charges = 2000
```

```python
gross_salary = base_salary + bonus
tax = gross_salary * tax_rate
net_salary = gross_salary - tax - other_charges
print(f" Gross Salary: ₹{gross_salary}")
print(f" Tax Deducted (10%): ₹{tax}")
print(f" Other Charges: ₹{other_charges}")
print(f" Net Salary: ₹{net_salary}")
```

```
Gross Salary: ₹55000
Tax Deducted (10%): ₹5500.0
Other Charges: ₹2000
Net Salary: ₹47500.0
```

In [13]:
```python
#6. Task: Bank Loan Approval System
#● Objective: You have to create a javascript script that checks whether an user is
#eligible for a bank loan based on various criteria.
#● Hints:
#○ The applicant's age must be between 18 and 60 years.
#○ The applicant's monthly income must be greater than or equal to ₹25000.
#○ The applicant must not have any outstanding debts greater than ₹10000
#1. Output:
#○ Display "Loan Approved" if the applicant meets all the conditions.
#○ Otherwise display "Loan Rejected".




age = int(input("Enter applicant's age: "))
income = float(input("Enter monthly income (₹): "))
credit_score = int(input("Enter credit score: "))
outstanding_debt = float(input("Enter outstanding debt (₹): "))

if (18 <= age <= 60 and
    income >= 25000 and
    credit_score >= 700 and
    outstanding_debt <= 10000):
    print(" Loan Approved")
else:
    print(" Loan Rejected")
```

```
Loan Approved
```

In [14]:
```python
#7. Task: Students Interview Eligibility Checker
#● Objective:you have to design a javascript script that checks whether a student i
#and extracurricular participation.
#● Input:
#○ Academic Score (percentage): A floating-point number representing the
#student's academic score. Ex .78.88
#○ Attendance Percentage: A floating-point number representing the
#student's attendance percentage. Ex.85.88
#○ Extracurricular Participation: This indicates whether the student has
#participated in any extracurricular activities. Ex.Yes/no

#Conditions for Interview Eligibility:
#1. The student's academic score must be 60 or above.
#2. The student's attendance percentage must be 75 or above.
#3. The student should have participated in at least one extracurricular activity.
```

```python
#Output:
#• If the student meets all three conditions print "Eligible for Interview".
#• If the student fails to meet any of the conditions print "Not Eligible for Inter




academic_score = float(input("Enter Academic Score (%): "))
attendance = float(input("Enter Attendance Percentage (%): "))
extracurricular = input("Participated in Extracurricular Activities? (Yes/No): ").s
if (academic_score >= 60 and
    attendance >= 75 and
    extracurricular == "yes"):
    print("Eligible for Interview")
else:
    print(" Not Eligible for Interview")
```

```
Eligible for Interview
```

In [16]:
```python
#8. Task: Validating Email Domain
#• Objective: You will implement a javascript program to validate the domain of a
#user's email address. The program will check if the email contains a specific
#domain (e.g. "gmail.com").
#Problem Statement:
#You are building a registration system that only accepts email addresses from a ce
#domain (e.g. "gmail.com"). Your task is to:

#1. Prompt the user to enter their email address.
#2. Check if the entered email address contains the domain "gmail.com".
#3. Display whether the email is eligible for registration based on the domain
#check.

#4. Print a message to inform the user if their email is eligible for registration
#or not.



email = input("Enter your email address: ").strip().lower()
if "@gmail.com" in email:
    print(" Email is eligible for registration.")
else:
    print(" Email is not eligible for registration.")
```

```
Email is eligible for registration.
```

In [3]:
```python
#Task 9: Employee Access Level System
#Objective:
#You are building a security system for a corporate office where employees are assi
#different access levels:
#• Admin Access
#• Staff Access
#• Visitor Access

#Scenario:
#The system must check whether two employees have been assigned exactly the same
#access level object in the system. This is important to avoid mistakenly duplicati
#access levels in memory or logic.
```

```python
#Instructions:

#1. ADMIN = "admin"
#2. STAFF = "staff"
#3. VISITOR = "visitor"
#• Assign access levels to two employees using these predefined constants.

#• Display:
#o "Both employees have the same access level object." if
#they point to the same object.

#o "Employees have different access level objects."
#otherwise.

ADMIN = "admin"
STAFF = "staff"
VISITOR = "visitor"
employee2_access = ADMIN
if employee1_access is employee2_access:
    print("Both employees have the same access level object.")
else:
    print("Employees have different access level objects.")
```

```
Both employees have the same access level object.
```

In [12]:
```python
#10.Task : Student Grading System
#Create a javascript program to calculate a student's grade based on their marks.
#Task:
#1. Input: Prompt the user to enter their marks.
#2. Criteria:
#o Grade A: 90-100
#o Grade B: 80-89
#o Grade C: 70-79
#o Grade D: 60-69
#o Grade E: 50-59
#o Grade F: 0-49
#o Invalid marks: Outside the range 0-100.
#3. Output: Display the grade or an error message for invalid marks.
#Example Outputs:
#• Marks: 85 → Grade: B
#• Marks: 45 → Grade: F
#• Marks: 105 → Invalid marks.


input_marks = input("Enter your marks (0-100): ").strip()
if input_marks == "":
    print("Input cannot be empty.")
else:
    try:
        marks = float(input_marks)
        if marks < 0 or marks > 100:
            print("Invalid marks. Please enter a value between 0 and 100.")
        elif marks >= 90:
            print("Grade: A")
        elif marks >= 80:
```

```python
            print("Grade: B")
        elif marks >= 70:
            print("Grade: C")
        elif marks >= 60:
            print("Grade: D")
        elif marks >= 50:
            print("Grade: E")
        else:
            print("Grade: F")
    except ValueError:
        print("Please enter a valid number.")
```

Grade: B

In [1]:
```python
#11.Task : Authentication System.
#Write a javascript program that authenticates a user by checking their username an
#password. The program should compare the entered credentials with predefined valid
#credentials.
#● Predefined valid usernames and corresponding passwords
#username1 = "user1"
#username1_password1 = "pass@123"

#Instructions:
#1. Input:
#o Prompt the user to input their username and password.
#2. Processing:
#o Check if the username and password match
#3. Output:
#o If both the username and password match the predefined valid credentials
#display "Authentication successful."
#o If either the username or the password does not match display
#"Authentication failed."


valid_username = "user1"
valid_password = "pass@123"
username = input("Enter your username: ").strip()
password = input("Enter your password: ").strip()
if username == valid_username and password == valid_password:
    print("Authentication successful.")
else:
    print("Authentication failed.")
```

Authentication successful.

In [2]:
```python
#12.Employee Salary Based on Experience.
#You are building a system for a Human Resources (HR) department that calculates an
#employee's salary based on their years of experience. The system assigns salary ti
#based on the number of years an employee has been working. It also offers bonuses
#employees who have more than 15 years of experience.
#Scenario Breakdown:
#Context 1: Senior Employee
#● An employee with 10 or more years of experience is classified as a Senior
#Employee. The base salary for such an employee is 80000.
#● If the employee has more than 15 years of experience they receive a bonus of
#5000 to their salary.
```

```python
#Example:
#An employee with 18 years of experience:
#● They are classified as Senior Employees.
#● Their base salary is 80000.
#● Since they have more than 15 years of experience they receive an additional
#5000 bonus.

#● The final salary is 85000.

#Context 2: Mid-Level Employee
#● Employees with 5 to 9 years of experience are classified as Mid-Level
#Employees.
#● Their base salary is 50000 with no bonus.

#Example:
#● They are classified as a Mid-Level Employee.
#● Their base salary is 50000.
#● Since they have fewer than 10 years of experience no bonus is added.
#● The final salary is 50000.

#Context 3: Junior Employee
#● Employees with less than 5 years of experience are classified as Junior
#Employees.
#● Their base salary is 30000 with no bonus.

#Example:
#An employee with 3 years of experience:
#● They are classified as Junior Employees.
#● Their base salary is 30000.
#● No bonus is added.
#● The final salary is 30000.

#Output Examples:
#Senior Employee with 18 years of experience:
#Enter years of experience: 18

#Senior employee
#Experience exceeds 15 years. Bonus added.
#Salary: 85000

#Mid-Level Employee with 7 years of experience:
#Enter years of experience: 7
#Mid-level employee
#Salary: 50000

#Junior Employee with 3 years of experience:
#Enter years of experience: 3
#Junior employee
#Salary: 30000


try:
    experience = int(input("Enter years of experience: "))
    if experience >= 10:
        print("Senior employee")
```

```python
        salary = 80000
        if experience > 15:
            print("Experience exceeds 15 years. Bonus added.")
            salary += 5000
    elif experience >= 5:
        print("Mid-level employee")
        salary = 50000
    elif experience >= 0:
        print("Junior employee")
        salary = 30000
    else:
        print("Invalid input. Experience cannot be negative.")
        salary = None
    if salary is not None:
        print("Salary:", salary)

except ValueError:
    print("Invalid input. Please enter a valid number for experience.")
```

```
Senior employee
Experience exceeds 15 years. Bonus added.
Salary: 85000
```

In [3]:
```python
try:
    experience = int(input("Enter years of experience: "))
    if experience >= 10:
        print("Senior employee")
        salary = 80000
        if experience > 15:
            print("Experience exceeds 15 years. Bonus added.")
            salary += 5000
    elif experience >= 5:
        print("Mid-level employee")
        salary = 50000
    elif experience >= 0:
        print("Junior employee")
        salary = 30000
    else:
        print("Invalid input. Experience cannot be negative.")
        salary = None
    if salary is not None:
        print("Salary:", salary)

except ValueError:
    print("Invalid input. Please enter a valid number for experience.")
```

```
Senior employee
Salary: 80000
```

In [4]:
```python
try:
    experience = int(input("Enter years of experience: "))
    if experience >= 10:
        print("Senior employee")
        salary = 80000
        if experience > 15:
            print("Experience exceeds 15 years. Bonus added.")
            salary += 5000
```

```python
        elif experience >= 5:
            print("Mid-level employee")
            salary = 50000
        elif experience >= 0:
            print("Junior employee")
            salary = 30000
        else:
            print("Invalid input. Experience cannot be negative.")
            salary = None
        if salary is not None:
            print("Salary:", salary)

    except ValueError:
        print("Invalid input. Please enter a valid number for experience.")
```

```
Mid-level employee
Salary: 50000
```

In [5]:
```python
try:
    experience = int(input("Enter years of experience: "))
    if experience >= 10:
        print("Senior employee")
        salary = 80000
        if experience > 15:
            print("Experience exceeds 15 years. Bonus added.")
            salary += 5000
    elif experience >= 5:
        print("Mid-level employee")
        salary = 50000
    elif experience >= 0:
        print("Junior employee")
        salary = 30000
    else:
        print("Invalid input. Experience cannot be negative.")
        salary = None
    if salary is not None:
        print("Salary:", salary)

except ValueError:
    print("Invalid input. Please enter a valid number for experience.")
```

```
Junior employee
Salary: 30000
```

In [1]:
```python
#13. Library Charge Calculation
#Problem Statement:
#Write a javascript program that calculates the library charge based on the number
#book has been borrowed.
#Charge Criteria:
#• Up to 5 days: Rs. 2 per day
#• 6 to 10 days: Rs. 3 per day
#• 11 to 15 days: Rs. 4 per day
#• More than 15 days: Rs. 5 per day
#Instructions:
#1. Input: Prompt the user to enter the number of days the book has been borrowed.
#2. Processing: Calculate the charge based on the given criteria.
#3. Output: Display the calculated charge.
```

```python
try:
    days = int(input("Enter number of days the book has been borrowed: "))
    if days <= 0:
        print("Invalid input. Days must be greater than 0.")
    elif days <= 5:
        charge = days * 2
    elif days <= 10:
        charge = days * 3
    elif days <= 15:
        charge = days * 4
    else:
        charge = days * 5
    if days > 0:
        print("Library charge: Rs.", charge)

except ValueError:
    print("Invalid input. Please enter a valid number of days.")
```

Library charge: Rs. 24

In [2]:
```python
try:
    days = int(input("Enter number of days the book has been borrowed: "))
    if days <= 0:
        print("Invalid input. Days must be greater than 0.")
    elif days <= 5:
        charge = days * 2
    elif days <= 10:
        charge = days * 3
    elif days <= 15:
        charge = days * 4
    else:
        charge = days * 5
    if days > 0:
        print("Library charge: Rs.", charge)

except ValueError:
    print("Invalid input. Please enter a valid number of days.")
```

Library charge: Rs. 30

In [3]:
```python
try:
    days = int(input("Enter number of days the book has been borrowed: "))
    if days <= 0:
        print("Invalid input. Days must be greater than 0.")
    elif days <= 5:
        charge = days * 2
    elif days <= 10:
        charge = days * 3
    elif days <= 15:
        charge = days * 4
    else:
        charge = days * 5
    if days > 0:
        print("Library charge: Rs.", charge)
```

```
    except ValueError:
        print("Invalid input. Please enter a valid number of days.")
```

Library charge: Rs. 100

In [1]:
```
#14. Arranging Three Numbers in Descending Order
#Task:
#Write a javascript program to arrange three numbers in descending order.

#Input:
#Prompt the user to enter three numbers.
#Processing:
#Sort the numbers in descending order.
#Example:
#• Enter first number: 3
#• Enter second number: 1
#• Enter third number: 2
#Output:
#• Numbers in Descending Order: 3, 2, 1

num1 = float(input("Enter first number: "))
num2 = float(input("Enter second number: "))
num3 = float(input("Enter third number: "))
numbers = [num1, num2, num3]
numbers.sort(reverse=True)
print("Numbers in Descending Order:", numbers[0], ",", numbers[1], ",", numbers[2])
```

Numbers in Descending Order: 3.0 , 2.0 , 1.0

In [1]:
```
#15. Tax Calculation for Car Purchase
#Write a program to calculate the tax on a car purchase based on the car brand and
#1. Mahindra: 5% tax for prices between 7L (7 lakh) and 10L.
#2. Audi: 10% tax for prices between 10L and 15L.
#3. Jaguar: 25% tax for prices between 15L and 20L.
#4. Mercedes: 30% tax for prices between 20L and 25L.
#5. Input: The car brand and price.
#6. Output: The calculated tax on the purchase.



brand = input("Enter car brand (Mahindra, Audi, Jaguar, Mercedes): ").strip().lower
price = float(input("Enter car price in lakhs (e.g., 12.5 for 12.5L): "))
tax = 0
if brand == "mahindra" and 7 <= price <= 10:
    tax = price * 0.05
elif brand == "audi" and 10 < price <= 15:
    tax = price * 0.10
elif brand == "jaguar" and 15 < price <= 20:
    tax = price * 0.25
elif brand == "mercedes" and 20 < price <= 25:
    tax = price * 0.30
else:
    print("No applicable tax for the given brand and price range.")
    tax = None
```

```python
    if tax is not None:
        print(f"Calculated tax on {brand.title()} purchase: Rs. {tax * 100000:.2f}")
```

No applicable tax for the given brand and price range.

In [2]:
```python
brand = input("Enter car brand (Mahindra, Audi, Jaguar, Mercedes): ").strip().lower
price = float(input("Enter car price in lakhs (e.g., 12.5 for 12.5L): "))
tax = 0
if brand == "mahindra" and 7 <= price <= 10:
    tax = price * 0.05
elif brand == "audi" and 10 < price <= 15:
    tax = price * 0.10
elif brand == "jaguar" and 15 < price <= 20:
    tax = price * 0.25
elif brand == "mercedes" and 20 < price <= 25:
    tax = price * 0.30
else:
    print("No applicable tax for the given brand and price range.")
    tax = None
if tax is not None:
    print(f"Calculated tax on {brand.title()} purchase: Rs. {tax * 100000:.2f}")
```

Calculated tax on Audi purchase: Rs. 150000.00

In [3]:
```python
brand = input("Enter car brand (Mahindra, Audi, Jaguar, Mercedes): ").strip().lower
price = float(input("Enter car price in lakhs (e.g., 12.5 for 12.5L): "))
tax = 0
if brand == "mahindra" and 7 <= price <= 10:
    tax = price * 0.05
elif brand == "audi" and 10 < price <= 15:
    tax = price * 0.10
elif brand == "jaguar" and 15 < price <= 20:
    tax = price * 0.25
elif brand == "mercedes" and 20 < price <= 25:
    tax = price * 0.30
else:
    print("No applicable tax for the given brand and price range.")
    tax = None
if tax is not None:
    print(f"Calculated tax on {brand.title()} purchase: Rs. {tax * 100000:.2f}")
```

Calculated tax on Jaguar purchase: Rs. 500000.00

In [4]:
```python
brand = input("Enter car brand (Mahindra, Audi, Jaguar, Mercedes): ").strip().lower
price = float(input("Enter car price in lakhs (e.g., 12.5 for 12.5L): "))
tax = 0
if brand == "mahindra" and 7 <= price <= 10:
    tax = price * 0.05
elif brand == "audi" and 10 < price <= 15:
    tax = price * 0.10
elif brand == "jaguar" and 15 < price <= 20:
    tax = price * 0.25
elif brand == "mercedes" and 20 < price <= 25:
    tax = price * 0.30
else:
    print("No applicable tax for the given brand and price range.")
    tax = None
```

```
        if tax is not None:
            print(f"Calculated tax on {brand.title()} purchase: Rs. {tax * 100000:.2f}")
```

Calculated tax on Mercedes purchase: Rs. 750000.00

In [5]:
```
#16.Finding the Middle Number

#○ Task: Write a program to determine the middle number among three inputs.
#○ Input: Prompt the user to enter three numbers.
#○ Processing: Identify the middle number, which is neither the largest nor the
#smallest.
#○ Output: Display the middle number.




a = float(input("Enter first number: "))
b = float(input("Enter second number: "))
c = float(input("Enter third number: "))
if (a > b and a < c) or (a < b and a > c):
    middle = a
elif (b > a and b < c) or (b < a and b > c):
    middle = b
else:
    middle = c
print("Middle number is:", middle)
```

Middle number is: 2.0

In [6]:
```
#17.Find the greatest number.

#○ Task: Write a program to find greatest number from three number
#○ Input: Prompt the user to enter three numbers.
#○ Output: Display the greatest number.




a = float(input("Enter first number: "))
b = float(input("Enter second number: "))
c = float(input("Enter third number: "))
greatest = max(a, b, c)
print("Greatest number is:", greatest)
```

Greatest number is: 70.0

In [7]:
```
#18.Authentication System

#○ Task: Write a program to authenticate a user by validating their username and
#password.
#○ Predefined Credentials:
#■ Username: user1
#■ Password: pass@123
#○ Input: Prompt the user to input their username and password.
#○ Output:
#■ If the credentials match, display "Authentication successful."
#■ If they do not match, display "Authentication failed."
```

```
correct_username = "user1"
correct_password = "pass@123"
username = input("Enter username: ").strip()
password = input("Enter password: ").strip()
if username == correct_username and password == correct_password:
    print("Authentication successful.")
else:
    print("Authentication failed.")
```

Authentication successful.

In [8]:
```
#19.Calculate Class Attendance Percentage

#○ Task: Write a program to calculate the percentage of classes attended by a
#student and determine their eligibility to sit in the exam.
#○ Conditions:
#■ Attendance percentage < 75%: Not eligible to sit in the exam.
#■ Attendance percentage ≥ 75%: Eligible to sit in the exam.
#○ Output: Display the attendance percentage and eligibility status.



total_classes = int(input("Enter total number of classes held: "))
attended_classes = int(input("Enter number of classes attended: "))
if total_classes > 0:
    attendance_percentage = (attended_classes / total_classes) * 100
    print(f"Attendance Percentage: {attendance_percentage:.2f}%")
    if attendance_percentage >= 75:
        print("Status: Eligible to sit in the exam.")
    else:
        print("Status: Not eligible to sit in the exam.")
else:
    print("Invalid input. Total classes must be greater than 0.")
```

Attendance Percentage: 87.50%
Status: Eligible to sit in the exam.

In [9]:
```
total_classes = int(input("Enter total number of classes held: "))
attended_classes = int(input("Enter number of classes attended: "))
if total_classes > 0:
    attendance_percentage = (attended_classes / total_classes) * 100
    print(f"Attendance Percentage: {attendance_percentage:.2f}%")
    if attendance_percentage >= 75:
        print("Status: Eligible to sit in the exam.")
    else:
        print("Status: Not eligible to sit in the exam.")
else:
    print("Invalid input. Total classes must be greater than 0.")
```

Attendance Percentage: 91.67%
Status: Eligible to sit in the exam.

In [11]:
```python
#20.Library Charge Calculation

#○ Task: Write a program to calculate the library charges based on the number of
#days a book has been borrowed.
#○ Charge Criteria:
#■ Up to 5 days: ₹2/day.
#■ 6 to 10 days: ₹3/day.
#■ 11 to 15 days: ₹4/day.
#■ More than 15 days: ₹5/day.
#○ Output: Display the total charges.


days = int(input("Enter number of days the book has been borrowed: "))
if days <= 0:
    print("Invalid input. Days must be greater than 0.")
elif days <= 5:
    charge = days * 2
elif days <= 10:
    charge = days * 3
elif days <= 15:
    charge = days * 4
else:
    charge = days * 5
if days > 0:
    print("Total Library Charges: ₹", charge)
```

Total Library Charges: ₹ 60

In [1]:
```python
#21.UPSC Selection Process

#○ Task: Simulate the UPSC selection process with the following steps:
#1. Eligibility Check
#■ Age: 21–32 years.
#■ Graduate status: Must be a graduate.
#■ Nationality: Must be "Indian".
#■ Output:
#■ If eligible, proceed to Prelims.
#■ If ineligible, display the reason for ineligibility.

#2. Prelims Exam
#■ Processing: Check if the candidate's score ≥ cut-off.
#■ Output:
#■ If passed, proceed to Mains.
#■ If failed, display "You failed the Prelims."

#3. Mains Exam
#■ Processing: Check if the candidate's score ≥ cut-off.
#■ Output:
#■ If passed, proceed to Interview.
#■ If failed, display "You failed the Mains."

#4. Interview
#■ Processing: Check if the candidate's score ≥ cut-off.
#■ Output:
#■ If passed, display "Congratulations! You have cleared the
#UPSC."
```

```python
#■ If failed, display "You failed the Interview."

#○ Final Output: Use nested conditional statements to simulate the entire process.



age = int(input("Enter your age: "))
graduate = input("Are you a graduate? (yes/no): ").strip().lower()
nationality = input("Enter your nationality: ").strip().lower()

if 21 <= age <= 32 and graduate == "yes" and nationality == "indian":
    print(" Eligible for UPSC. Proceed to Prelims.")


    prelims_score = float(input("Enter Prelims score: "))
    prelims_cutoff = float(input("Enter Prelims cut-off: "))

    if prelims_score >= prelims_cutoff:
        print(" Passed Prelims. Proceed to Mains.")


        mains_score = float(input("Enter Mains score: "))
        mains_cutoff = float(input("Enter Mains cut-off: "))

        if mains_score >= mains_cutoff:
            print(" Passed Mains. Proceed to Interview.")


            interview_score = float(input("Enter Interview score: "))
            interview_cutoff = float(input("Enter Interview cut-off: "))  # ✅ Fix

            if interview_score >= interview_cutoff:
                print(" Congratulations! You have cleared the UPSC.")
            else:
                print(" You failed the Interview.")
        else:
            print(" You failed the Mains.")
    else:
        print(" You failed the Prelims.")
else:
    print(" Not eligible for UPSC.")
    if not (21 <= age <= 32):
        print("Reason: Age must be between 21 and 32.")
    if graduate != "yes":
        print("Reason: Must be a graduate.")
    if nationality != "indian":
        print("Reason: Nationality must be Indian.")
```

```
Eligible for UPSC. Proceed to Prelims.
Passed Prelims. Proceed to Mains.
Passed Mains. Proceed to Interview.
Congratulations! You have cleared the UPSC.
```

In [2]:
```python
#22. Menu-Driven Login System
#1. Create the Menu:
#o Display a menu with three choices for the user:
#■ Login with Phone
#■ Login with Email

#■ Exit the system
#2. Predefined Credentials:
#o Phone number: "1234567890"
#o OTP: "1234"
#o Email: "user@example.com"
#o Password: "password123"
#3. Login Functionality:
#o Option 1 (Login with Phone):
#■ Prompt the user to enter their phone number and OTP.
#■ Compare the input with a predefined phone number and OTP.
#■ Display success if both match or an error message if they don't.
#o Option 2 (Login with Email):
#■ Prompt the user to enter their email and password.
#■ Compare the input with predefined email and password.
#■ Display success if both match or an error message if they don't.
#o Option 3 (Exit):
#■ Display an exit message and terminate the program.
#■ Handle invalid user choices and ask the user to select a valid option.

#Output:
#1. If the user enters a valid phone number and OTP, display: "Login successful
#with phone!"
#2. If the user enters valid email and password, display: "Login successful
#with email!"
#3. If the user selects the exit option, display: "Exiting the program. Have a
#nice day!"
#4. If the user enters invalid credentials or an invalid choice, display appropriat
#messages.


phone_number = "1234567890"
otp = "1234"
email = "user@example.com"
password = "password123"

while True:
    print("\n--- UPSC Login Menu ---")
    print("1. Login with Phone")
    print("2. Login with Email")
    print("3. Exit")

    choice = input("Enter your choice (1/2/3): ").strip()

    if choice == "1":
        entered_phone = input("Enter your phone number: ").strip()
        entered_otp = input("Enter OTP: ").strip()
```

```python
            if entered_phone == phone_number and entered_otp == otp:
                print(" Login successful with phone!")
            else:
                print(" Invalid phone number or OTP.")

        elif choice == "2":
            entered_email = input("Enter your email: ").strip().lower()
            entered_password = input("Enter your password: ").strip()

            if entered_email == email and entered_password == password:
                print(" Login successful with email!")
            else:
                print(" Invalid email or password.")

        elif choice == "3":
            print(" Exiting the program. Have a nice day!")
            break

        else:
            print(" Invalid choice. Please select 1, 2, or 3.")
```

```
--- UPSC Login Menu ---
1. Login with Phone
2. Login with Email
3. Exit
 Invalid phone number or OTP.

--- UPSC Login Menu ---
1. Login with Phone
2. Login with Email
3. Exit
 Invalid email or password.

--- UPSC Login Menu ---
1. Login with Phone
2. Login with Email
3. Exit
 Exiting the program. Have a nice day!
```

In [4]:
```python
#23.Create Your Own KBC Game
#Design and implement a quiz game inspired by the popular Kaun Banega Crorepati (KB
#game show. The aim of this assignment is to test the user's knowledge through a se
#multiple-choice questions, track their score, and display statistics at the end of
#game also provides the flexibility to skip any question.
#Instructions:

#1. Game Structure:
#o The game will consist of 5 multiple-choice questions.
#o The user will be asked a question with 4 options (A, B, C, D).
#o The user can choose to skip any question they do not want to answer.
#2. Scoring System:
#o Points will be awarded for correct answers as follows:
#■ Question 1 → 1000 points
#■ Question 2 → 2000 points
#■ Question 3 → 3000 points
#■ Question 4 → 5000 points
```

```python
#■ Question 5 → 10000 points
#○ For incorrect answers, no points will be awarded.
#○ For skipped questions, no points will be awarded, but the game will continue.
#3. End of Game Statistics:
#○ At the end of the game, the following statistics will be displayed:
#■ Total score accumulated from correct answers.
#■ Number of skipped questions.
#■ Number of wrong answers

#4. User Experience:
#○ At the beginning of the game, ask the user whether they would like to start or
#not the game.
#○ Provide the option for the user to skip any question at any point..
#5. Game Ending:
#○ The game will end when all the questions have been answered or skipped. The
#user should receive their total score and a summary of their performance.




questions = [
    {
        "question": "1. What is the capital of India?",
        "options": {"A": "Mumbai", "B": "Delhi", "C": "Kolkata", "D": "Chennai"},
        "answer": "B",
        "points": 1000
    },
    {
        "question": "2. Who wrote the national anthem of India?",
        "options": {"A": "Rabindranath Tagore", "B": "Bankim Chandra", "C": "Saroji
        "answer": "A",
        "points": 2000
    },
    {
        "question": "3. What is the chemical symbol for Gold?",
        "options": {"A": "Ag", "B": "Au", "C": "Gd", "D": "Go"},
        "answer": "B",
        "points": 3000
    },
    {
        "question": "4. Which planet is known as the Red Planet?",
        "options": {"A": "Earth", "B": "Venus", "C": "Mars", "D": "Jupiter"},
        "answer": "C",
        "points": 5000
    },
    {
        "question": "5. Who is known as the Missile Man of India?",
        "options": {"A": "Ratan Tata", "B": "A.P.J. Abdul Kalam", "C": "Vikram Sara
        "answer": "B",
        "points": 10000
    }
]


start = input(" Welcome to KBC! Do you want to start the game? (yes/no): ").strip()
if start != "yes":
```

```python
        print(" Exiting the game. See you next time!")
else:
    total_score = 0
    correct_answers = 0
    skipped_questions = 0
    wrong_answers = 0

    for q in questions:
        print("\n" + q["question"])
        for key, value in q["options"].items():
            print(f"  {key}. {value}")
        user_input = input("Enter your answer (A/B/C/D) or type 'skip' to skip: ").

        if user_input == "SKIP":
            print(" Question skipped.")
            skipped_questions += 1
        elif user_input == q["answer"]:
            print(" Correct!")
            total_score += q["points"]
            correct_answers += 1
        else:
            print(" Wrong answer.")
            wrong_answers += 1

    # Final Statistics
    print("\n Game Over! Here's your performance summary:")
    print(f"Total Score: {total_score}")
    print(f"Correct Answers: {correct_answers}")
    print(f"Skipped Questions: {skipped_questions}")
    print(f"Wrong Answers: {wrong_answers}")
```

```
1. What is the capital of India?
  A. Mumbai
  B. Delhi
  C. Kolkata
  D. Chennai
 Correct!

2. Who wrote the national anthem of India?
  A. Rabindranath Tagore
  B. Bankim Chandra
  C. Sarojini Naidu
  D. Jawaharlal Nehru
 Correct!

3. What is the chemical symbol for Gold?
  A. Ag
  B. Au
  C. Gd
  D. Go
```

```
 Correct!

4. Which planet is known as the Red Planet?
  A. Earth
  B. Venus
  C. Mars
  D. Jupiter
 Correct!

5. Who is known as the Missile Man of India?
  A. Ratan Tata
  B. A.P.J. Abdul Kalam
  C. Vikram Sarabhai
  D. Homi Bhabha
 Correct!

 Game Over! Here's your performance summary:
Total Score: 21000
Correct Answers: 5
Skipped Questions: 0
Wrong Answers: 0
```

In [ ]: